AFRL-AFOSR-JP-TR-2016-0044

Activity Recognition in Social Media

Subhasis Chaudhuri
INDIAN INSTITUTE OF TECHNOLOGY BOMBAY

05/09/2016
Final Report

Air Force Research Laboratory
AF Office Of Scientific Research (AFOSR)/ IOA
Arlington, Virginia 22203
Air Force Materiel Command

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
09-05-2016

2. REPORT TYPE
Final

3. DATES COVERED *(From - To)*
12 Aug 2013 to 30 Sep 2015

4. TITLE AND SUBTITLE
Activity Recognition in Social Media

5a. CONTRACT NUMBER
FA2386-13-1-4139

5b. GRANT NUMBER
Grant 12RSZ100_134139

5c. PROGRAM ELEMENT NUMBER
61102F

6. AUTHOR(S)
Subhasis Chaudhuri

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
INDIAN INSTITUTE OF TECHNOLOGY BOMBAY
POWAI
MUMBAI, 400076 IN

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AOARD
UNIT 45002
APO AP 96338-5002

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/AFOSR/IOA(AOARD)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
12RSZ100_134139

12. DISTRIBUTION/AVAILABILITY STATEMENT
Distribution Code A: Approved for public release, distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
A novel approach to analyze crowd behavior at various levels of granularity - individual, group and global. We first model the collective motion of the agents present in the scene by a first order dynamical system. The model learns the spatio-temporal interaction pattern of the crowd which is further analyzed for group detection. The groups are identiable from the eigenvectors of the interaction matrix of the model and can be recovered by employing a variant of spectral clustering on the eigenvectors. We show that while eigenvectors detect groups, the eigenvalues characterize various group activities such as stationary, walking, splitting and approaching. Finally we classify a crowd video in one of the eight categories by employing a random forest. As an application, the model is used to predict personal space violation.

15. SUBJECT TERMS
Social Media, AOARD, social interactions

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	c. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES
Unclassified	Unclassified	Unclassified	SAR	15

19a. NAME OF RESPONSIBLE PERSON
HONG, SENG

19b. TELEPHONE NUMBER *(Include area code)*
315-229-3519

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

DISTRIBUTION A: Distribution approved for public release

Final report for AOARD Grant FA23861314139

Activity Recognition in Social Media

December 29, 2015

Name of Principal Investigator (PI): Prof. Subhasis Chaudhuri

- **E-mail address**: sc@ee.iitb.ac.in
- **Institution**: Indian Institute of Technology Bombay, India
- **Phone**: (91-22) 2576 7010

Period of Performance:

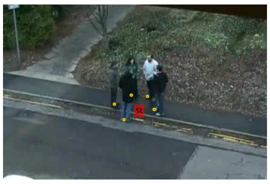
Abstract

In this work, we present a novel approach to analyze crowd behavior at various levels of granularity – individual, group and global. We first model the collective motion of the agents present in the scene by a first order dynamical system. The model learns the spatio-temporal interaction pattern of the crowd which is further analyzed for group detection. The groups are identifiable from the eigenvectors of the interaction matrix of the model and can be recovered by employing a variant of spectral clustering on the eigenvectors. We show that while eigenvectors detect groups, the eigenvalues characterize various group activities such as stationary, walking, splitting and approaching. Finally we classify a crowd video in one of the eight categories by employing a random forest. As an application, the model is used to predict personal space violation.

1 Introduction

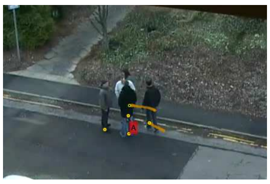
Understanding human behavior at an individual level, at a group level and at a crowd level in different scenarios has always attracted the researchers. The variability and complexity in the behavior make it a highly challenging task. However, this decade is witnessing a huge interest of researchers in the area of crowd motion analysis due to its various applications in surveillance, safety, public place management, hazards prevention, and virtual environments. This interest has resulted in many interesting papers in the area.

We are aware of at least four survey papers on the subject of crowd analysis that indicate the amount of attention, it has drawn in this and the previous decade [5],[9],[4],[10]. The latest survey paper [5] by Chang *et al.* encapsulates the recent works published after 2009, covering topics of motion pattern segmentation, crowd behavior and anomaly detection. Thida *et al.* [9] provide a review on macroscopic and microscopic modeling methods. They also present a critical survey on crowd event detection. Julio *et al.* cover various vision techniques applicable to crowd analysis such as tracking, density estimation, and computer simulation [4]. Zhan *et al.* discuss various vision based techniques used in crowd analysis. They also discuss crowd analysis from the perspective of different disciplines - psychology, sociology and computer graphics [10]. At the top level, the techniques used in crowd motion analysis can be divided into two major classes $-$ holistic and particle based. The holistic methods consider crowd as a single entity and analyze the overall behavior. These methods fail to provide much insight at an individual or intermediate level. On the other hand, particle based methods consider crowd as a collection of individuals or groups. But their performance degrades with the increase in crowd density due to occlusion and tracking problems.



(a) Stationary group

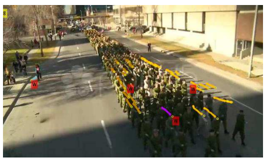
(b) Walking



(c) Approaching

(d) Splitting

(e) Mixed crowd



(f) Uniform crowd

Figure 1: (a) - (d) show groups with different group activities, (e) and (f) give examples of structured and unstructured crowd. Tracklets for some of the agents over past few frames are also shown. Each color represents a group (Best viewed in color). The videos are from BEHAVE [1] and CUHK [8] datasets.

We believe that a moderately dense crowd consists of various groups. We define a group as a set of individuals having some sort of interaction. Spatial proximity is required to form a group; if there are agents with a similar motion pattern but are far away from each other, they do not form a group as per our definition. Each group has its own set of goals that leads to various interaction patterns among the members of the group. The collective behavior of these constituent groups identifies the global crowd behavior which can vary from a highly structured to a totally unstructured pattern. In case of a structured crowd, for example − marching of soldiers, all groups are in coordination and share the same goal (see Fig.1f); whereas in an unstructured crowd, for example − at railway station or at a shopping complex, there are multiple groups with different goals (see Fig.1e). We are interested in understanding these different types of crowd behaviors at various levels.

2 Mathematical formulation

We define a group as a set of agents having spatial proximity and some sort of interaction. In general, such interactions are complex and non-linear in nature. We approximate these interactions locally in time by a first order dynamical model. Note that we refer by agent an individual entity in the crowd.

2.1 Proposed interaction model

We model the collective relationship among the agents by a first order homogeneous system. Our hypothesis is based on the intuition that each agent takes into consideration (*i*) the movement of other agents present nearby and (*ii*) her/his desired goal, while taking the next step. The model relates the next positions of the agents to the current positions. Let $\mathbf{x}(k) = [x_1(k), x_2(k), ..., x_N(k)]^T$, then

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k), \tag{1}$$

where $\mathbf{A} \in \mathbb{R}^{N \times N}$, N is the total number of agents and $x_i(k) \in \mathbb{R}$ is the location of i^{th} agent at time instant k along x-axis. We call $\mathbf{A}$ as the interaction matrix which captures the evolution of an agent as a function of all agents present in the scene. Note that $\mathbf{A}$ has no assumption on its form and entries. It need not be symmetric i.e. agent i may not depend on agent j in the same way as agent j depends on agent. For example, consider a case where agent i is stationary and agent j approaches him/her. Since their behaviors are not symmetric with respect to each other, intuitively that means $a_{ij} \neq a_{ji}$.

In this paper, it is assumed that the motion along x and y directions are independent and hence can be analyzed independently. The corresponding model along y direction is $\mathbf{y}(k+1) = \mathbf{B}\mathbf{y}(k)$. In the rest of the paper, we discuss the solution for matrix $\mathbf{A}$ noting this fact and the same process is also carried out for $\mathbf{B}$. We expect matrices $\mathbf{A}$ and $\mathbf{B}$ to be dependent on crowd motion. Since crowd behavior might change with time, the interaction matrix is time varying in nature, that is $\mathbf{A}_k$. Assuming $\mathbf{A}$ has N independent eigenvectors, the general solution to Eq.(1) is given as

$$\mathbf{x}(k) = \sum_{i=1}^{N} c_i \lambda_i^k \mathbf{v}_i, \tag{2}$$

where λ_i is the i^{th} eigenvalue, $\mathbf{v}_i$ is the corresponding normalized eigenvector and c_i is the corresponding constant coefficient that depends on the initial condition. Different values of λ_i and $\mathbf{v}_i$ generate various motion patterns for an agent. These patterns can be associated to different motion tracks generated by an agent while walking, approaching, splitting or stationary.

2.2 Estimation of interaction matrix A

The matrix $\mathbf{A}$ at any time instant is learned from the immediate past trajectory data of all the agents in a least squares framework. We update $\mathbf{A}$ with each incoming frame as interaction patterns may change over the time. In addition, sudden changes in these interactions are unlikely. Therefore it is desired that the entires of $\mathbf{A}$ do not change drastically in consecutive time instants $-$ we assume them to be varying smoothly over time. We incorporate this constraint by minimizing l_2 norm of the difference between current interaction matrix $\mathbf{A}_k$ and previous estimate at $(k-1)^{th}$ instant . Furthermore for crowded scenes, it is unlikely that an agent's motion depends on all the agents present in the scene. We capture sparsity in $\mathbf{A}_k$ by minimizing l_1 norm of $\mathbf{A}_k$. Adding these constraints to the cost function, the final formulation at k^{th} time instant becomes:

$$\begin{aligned} \mathbf{A}_k^* = \quad & \arg \min_{\mathbf{A}_k \in \mathbf{R}^{N\times N}} \Big\{ ||\mathbf{A}_k \mathbf{X}_{k-m}^{k-1} - \mathbf{X}_{k-m+1}^{k}||_2^2 \\ & +\lambda_1 ||\mathbf{A}_k - \mathbf{A}_{k-1}||_2^2 + \lambda_2 ||\mathbf{A_k}||_1 \Big\}, \end{aligned} \tag{3}$$

where $\mathbf{X}_i^j \in \mathbb{R}^{N\times m}$ contains the positions of all N agents from i^{th} to j^{th} frames concatenated together, $\mathbf{A}_{k-1}$ is the estimate at the previous frame and λ_1 and λ_2 are appropriate regularization parameters. Note that we will use $\mathbf{A}$ instead of $\mathbf{A}_k$ for notational convenience.

We use $m = 2.5N$ past positions to solve this optimization problem. Therefore the interaction pattern is assumed to remain constant over $2.5N$.

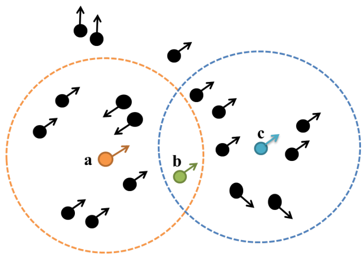


Figure 2: Spatial neighborhoods around agents **a** and **c** are represented as circles around them. There are a total of 20 agents in the scene out of which only 8 are neighbors of **a**. Estimation of elements of row of **A** corresponding to agent **a**, considering all agents present in the scene requires $2.5 * 20 = 50$ previous video frames. While the use of neighborhood constraint reduces this to $2.5 * 9 \approx 23$ frames.

However, a large N leads to two major problems: (i) longer trajectories are required to learn the interaction matrix and (ii) the interaction may not remain constant over $2.5N$ past positions. To address these problems, we identify spatial neighbors of each agent and learn only the corresponding entries in the matrix, others being zero. The neighborhood is defined as follows $-$ the agent $\mathbf{a}$ is a neighbor to the agent $\mathbf{b}$ if $dist(\mathbf{a}, \mathbf{b}) < R_{\mathbf{b}}$. The intuition is that it is unlikely that far away agents influence the motion of an agent. The advantage is that the shorter trajectories are now sufficient as the number of entries of **A** to be learned are lesser. Further, there could be an agent within the spatial proximity of another agent but there may not be any interaction between them. Hence it is required that the corresponding entry in the matrix **A** should be zero. This is enforced by adding sparsity constraint in Eq. 3. In essence, spatial proximity is taken into consideration by employing neighborhood based selection while temporal proximity is achieved by Eq. 3.

For an illustration, refer Fig.2. There are total of 20 agents present in the scene. Estimation of row of matrix **A** corresponding to agent **a** requires 50 previous frames whereas neighborhood based estimation reduces this to 23. Also consider a case where agents **a** and **c** are not in spatial proximity of each other but interact via agent **b** (for that matter, a chain of other agents), then it is captured by the row of **A** corresponding to agent **b** since it has both the agents in its neighborhood. Note that we estimate matrix **A** in a row-wise manner since the number of entries to be estimated is different for each agent due to neighborhood constraint. We use L1General package developed by Schmidt [7] for solving L1-regularization problems.

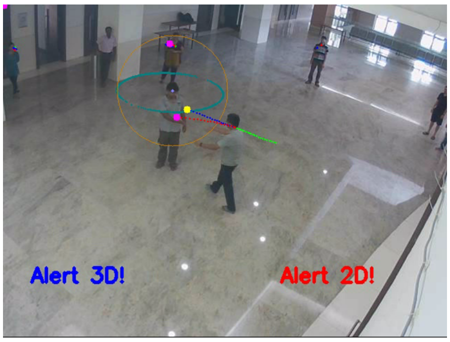


Figure 3: The figure compares the prediction of personal space in 2D and 3D using the proposed model. Best viewed in color

3 Prediction of personal space violation

The model is used to predict the future locations of the agents and hence can be used for predicting personal space violation. Firstly, human detection is done using deformable part based model [3] and each individual is represented by a point. These points are tracked using Lukas-Kanade tracker. The videos for this experiment were captured in the campus premises. Since the camera was calibrated and same height was assumed for all the agents, the 3D coordinates were estimated. The parameters of the motion matrix (discussed before) are estimated continuously. The trajectories are predicted and analysis is done for predicting personal space prediction. See Fig 3 for an example.

4 Group detection

In this section, we discuss the algorithm for identifying the groups presnt in the scene by analyzing the interaction matrix $\mathbf{A}$. From Eq. 2, notice that if any two rows of eigenvector matrix are similar, the corresponding agents belong to same group. Hence we define a mapping for i^{th} agent as $f(x_i) : x_i \in \mathbb{R} \rightarrow \mathbf{z}_i = (v_{1i}, v_{2i}, \ldots, v_{ri})^T \in \mathbb{R}^{r\times 1}$ where v_{ji} is the i^{th} entry of j^{th} eigenvector of interaction matrix $\mathbf{A}$ and r is the number of significant eigenvectors. A clustering algorithm is applied on the points $\{\mathbf{z}_i\}, \forall i = 1, 2, \ldots, N$. Since the clustering algorithm runs on the components of eigenvectors, this algorithm falls in the category of spectral clustering [6]. The number of groups in unknown, so we apply a threshold based clustering. The adaptive threshold used for i^{th} point is $c|\mathbf{z}_i|$, where c is found empirically. Also we consider only significant eigenvectors (with $|\lambda| \geq 0.90$, which was found empirically) of $\mathbf{A}$ for group detection since the response from the eigenvectors with $|\lambda| < 0.9$ dies down to an insignificant level immediately.

5 Group activity identification

While the eigenvectors identify the groups, the eigenvalues determine the activity of a group. We employ the same model mentioned in Eq. 1 for the group G to estimate its interaction matrix $\mathbf{A}^G$. We do not use the submatrix formed by the agents of the group G in the previously learned matrix $\mathbf{A}$ to get $\mathbf{A}^G$. This is to avoid any possible interference from the outside agents in the estimation and get a refined matrix for the group. Let $\mathbf{x}^G(k) = [x_1^G(k), x_2^G(k), \ldots, x_M^G(k)]^T$, where M is the cardinality of the group G and $x_i^G(k)$ is the position of i^{th} agent of the group at time instant k. To learn matrix $\mathbf{A}^G$ at k^{th} time instant, we define a similar optimization framework as follows, where the second term enforces temporal continuity in the activity but unlike Eq. 3, there is no need for sparsity constraint. Therefore,

$$\begin{aligned}\mathbf{A}_k^{G*} = \; & \arg \min_{\mathbf{A}_k^G \in \mathbf{R}^{M \times M}} \Big\{ ||\mathbf{A}_k^G \mathbf{X}_{k-m}^{k-1} - \mathbf{X}_{k-m+1}^{k}||_2^2 \\ & \qquad + \lambda ||\mathbf{A}_k^G - \mathbf{A}_{k-1}^G||_2^2 \Big\} \end{aligned} \tag{4}$$

Assuming $\mathbf{A}^G$ to be again diagonalizable, the general solution is

$$\mathbf{x}^G(k) = \sum_{i=1}^{M} d_i \lambda_i'^k \mathbf{u}_i, \tag{5}$$

where $|\lambda_1'| \geq |\lambda_2'| \ldots \geq |\lambda_M'|$. Now we state how eigenvalues determine various activities:

1. **Stationary**: A group is stationary when $|\lambda_i'| \in \{0, 1\}$, $\forall i$. To cater for the noisy measurements, we keep a positive threshold i.e. if $|\lambda_i'| < \theta$ (say, $\theta = 0.6$), the group is stationary.

2. **Walking**: Agents are walking or running together if $|\lambda_1'| > 1$ and their corresponding entries in $\mathbf{u}_1$ are closer. The fact that $\lambda_1' > 1$, corresponds to walking or running. The other fact that $\mathbf{u}_1$ has similar values suggests that agents are together.

3. **Approaching**: A few or all the agents of the group are approaching to meet if $\lambda_i' = 1$ and $0 < \lambda_j' \leq 1$, $\forall i \neq j$. The eigenvectors corresponding to $\lambda' = 1$ indicate the final location of meeting and $0 < \lambda' < 1$ suggest approaching behavior.

4. **Splitting**: A few or all the agents are splitting away from the group if $|\lambda_1'| > 1$ and $\mathbf{u}_1$ has different values corresponding to splitting agents.

This group activity detection method is highly dependent on eigenvalues and hence sensitive to perturbations in the measurements. To address this, we define threshold bands for crucial values of eigenvalues. For example, if $0.995 < \lambda < 1.005$, we consider λ to be 1 and so forth.

5.1 Atomic activity detection

This algorithm is extendable for identification of individual's activity. For an individual, we use the following model. Note that there is no longer a activity called splitting as one needs at least two agents to define it.

$$x(k+1) = \lambda x(k) + b \tag{6}$$

The solution is as follows:

$$x(k) = \begin{cases} \lambda^k x(0) + \frac{1-\lambda^k}{1-\lambda} b, & \text{if } \lambda \neq 1 \\ x(0) + kb, & \text{if } \lambda = 1 \end{cases} \tag{7}$$

We identify following activities based on the value of λ:

1. **Stationary**: An agent is stationary if $\lambda = 0$ at the location given by b.
2. **Approaching**: $0 < |\lambda| < 1$ indicates that the agent is approaching to the location b.
3. **Walking**: An agent is walking away from a reference point if $|\lambda| \geq 1$.

Note that the group detection and activity recognition algorithms run in x and y directions independently and results are combined together. For example if $L_x = [1, 1, 2]$ and $L_y = [1, 2, 1]$ are the label vectors (indicating groups) obtained in x and y directions respectively, the final label vector is $L = [1, 2, 3]$. To identify the final group activity from the estimates along x and y, we follow this priority sequence $-$ *Splitting* $>$ *Walking* $>$ *Approaching* $>$ *Stationary.* That is if a group has *splitting* and *approaching* activities in x and y directions respectively, the final group activity is *splitting.*

6 Crowd video classification

Ability to identify crowd behavior enables crowd management systems to design and manage public places effectively to ensure safety and smooth operation. The overall crowd behavior is determined by how each group behaves. Depending on the synchronization among the groups, the behavior of crowd varies from being structured to unstructured. In this section, we define group level features that are useful for crowd video classification. We classify crowd videos into 8 classes as defined by [8]. The dataset containing 474 video clips covers a variety of videos. The eight classes are as follows:

C1 : Mixed crowd
C2 : Well organized crowd following mainstream:
C3 : Not well organized crowd following any mainstream
C4 : Crowd merge
C5 : Crowd split
C6 : Crowd crossing in opposite directions
C7 : Intervened escalator traffic
C8 : Smooth escalator traffic

We employ group level features that cover low-level details such as motion information to high-level information such as group activities. The features are described as follows:

1. **Group density** (GD): It is the ratio of number of groups by the total number of agents in the scene. The low value of GD indicates highly structured crowd. For example, GD for a group of marching soldiers is small whereas a mixed crowd has a higher group density.

2. **Histogram of** λ_{max}: The histogram has three bins $-$ $\lambda_{max} > 1$, $\lambda_{max} = 1$ and $\lambda_{max} < 1$, where λ_{max} is the largest eigenvalue of the interaction matrix for a group. The value at a particular bin is the number of groups in a video clip having λ_{max} as defined by that bin. Left skewed histogram i.e. towards $\lambda_{max} > 1$ indicates moving crowd whereas right skewed histogram suggests more or less stationary crowd.

3. **Histogram of direction**: The motion direction of each member of a group is calculated from its trajectory data and the mean direction is assigned to the group. This histogram has eight bins covering $0°$ to $360°$ with a bin size of $45°$. The bin value is the number of groups falling in that particular bin. The uniform histogram indicates mixed crowd whereas skewed histogram indicates directional uniformity in the crowd.

4. **Histogram of group activity**: This is an important feature in deciding the overall activity. The histogram has 4 bins $-$ walk, stationary, approach, and split. The bin value is the number of groups performing the particular activity in the scene.

Since the analysis is conducted independently in x and y directions; we get two histograms for λ_{max}, leading to final feature vector of length $1+2\times 3+8+4 = 19$. We use random forest (RF) as a classifier [2]. It consists of multitude of decision trees that are trained from randomly sampled subsets of training dataset (bootstrap aggregating). This bootstrapping increases the performance by reducing the variance of the classifier. Also the split at each node of a tree is decided by m features selected randomly out of n features where $m << n$. We use RF to classify a crowd video by training it with the above mentioned features. The classification results are discussed in next section.

7 Experiments and Results

We tested our algorithms on BEHAVE [1] and CUHK datasets [8] which are quite common among the researchers for crowd analysis and group activity detection . CUHK dataset is a comprehensive crowd video dataset containing 474 video clips covering various crowd behaviors with varying crowd density. BEHAVE dataset has video clips covering various types of group activities.

7.1 Group detection

We tested group detection algorithm on 75 videos (covering all the different scenarios) from CUHK dataset and 2 video clips (having duration of more than 10 minutes in total) from BEHAVE dataset. We have excluded the clips containing other activities such as fight. In case of videos from CUHK dataset, we restricted our algorithm to run on around 60 longest tracks, since some of the clips are too short to accommodate for an analysis of large number of agents. We compared the proposed algorithm with other methods on these selected agents. The ground truth for CUHK dataset was obtained manually. Fig. 4 demonstrates a visual comparison for different scenarios

7.2 Crowd video classification

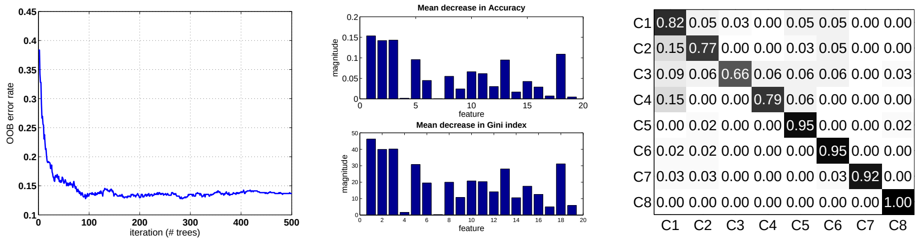
Since we update the interaction model with each incoming frame as explained in Section 6, we compute group level features at every time instant. We collect features at regular intervals from the videos to create the feature database. From each class, we randomly pick 70% feature vectors to train the classifier and the remaining for testing. As discussed before, we use random forest as a classifier with $n = 17$ and $m = 4$. We run the classifier 100 times with random splits of dataset for training and testing. The average accuracy obtained is 88%, a significant improvement over [8] where the reported accuracy is 70%. The confusion matrix is shown in Fig. 5c. The OOB error, which indicates generalized error, converges to a value less than 15% as shown in Fig. 5a. The importance plots, which show significance of each group level feature in the classification are shown in 5b.

8 Conclusions

In this work, we presented a framework for analysis of medium dense crowd videos at various levels. We proposed a first order dynamical system to

(a) Proposed (b) Ground truth

Figure 4: Comparison of group detection results from our proposed method in column (a) with the ground truth in column (b) for different types of scenes. Videos are from CUHK dataset [8]. Best viewed in color.



(a) (b) (c)

Figure 5: (a) Out of bag (OOB) error, (b) Importance plot for the features and (c) Confusion matrix with categories represented as C1 to C8.

model agent trajectories collectively and subsequently demonstrated the effectiveness of this interaction model for group detection. We also show how eigenvalues of the model characterize group activities. We then showed the effectiveness of group level features in crowd video classification.

9 List of Publications

1. N. Bhargava, S. Chaudhuri, and G. Seetharaman, “Linear cyclic pursuit based prediction of personal space violation in surveillance videos”, Proc. Applied Imagery Pattern Recognition Workshop (AIPR 2013), Washington DC, Oct 2013.

2. N. Bhargava and S. Chaudhuri, ”Finding group interactions in social gathering videos”, Proc. Indian Conf. on Vision, Graphics and Image Processing (ICVGIP), Bangalore, Dec 2014.

3. K. Vishal, N. Bhargava, S. Chaudhuri, and G. Seetharaman, ”Fast compensation of illumination changes for background subtraction”, Proc. Applied Imagery Pattern Recognition Workshop (AIPR 2013), Washington DC, Oct 2013.

4. N. Bhargava, S. Chaudhuri, ”An unified approach to crowd analysis using spatio-temporal interaction model”, Under preparartion.

References

[1] S. Blunsden and R. Fisher. The behave video dataset: ground truthed video for multi-person behavior classification. *Annals of the BMVA*, 4(1-12):4, 2010.

[2] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[3] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645, 2010.

[4] J. S. J. Junior, S. Musse, and C. Jung. Crowd analysis using computer vision techniques. *IEEE Signal Processing Magazine*, 5(27):66–77, 2010.

[5] T. Li, H. Chang, M. Wang, B. Ni, R. Hong, and S. Yan. Crowded scene analysis: A survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(3):367–386, 2015.

[6] A. Y. Ng, M. I. Jordan, Y. Weiss, et al. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 2:849–856, 2002.

[7] M. Schmidt, G. Fung, and R. Rosales. Optimization methods for l1-regularization. *University of British Columbia, Technical Report TR-2009*, 19, 2009.

[8] J. Shao, C. C. Loy, and X. Wang. Scene-independent group profiling in crowd. In *CVPR, 2014*, pages 2227–2234. IEEE, 2014.

[9] M. Thida, Y. L. Yong, P. Climent-Pérez, H.-l. Eng, and P. Remagnino. A literature review on video analytics of crowded scenes. In *Intelligent Multimedia Surveillance*, pages 17–36. Springer, 2013.

[10] B. Zhan, D. N. Monekosso, P. Remagnino, S. A. Velastin, and L.-Q. Xu. Crowd analysis: a survey. *Machine Vision and Applications*, 19(5-6):345–357, 2008.